

- IEEE/ACM International Conference on Automated Software Engineering*, pp. 1129–1140, 2020.
- [23] F. N. Naseem, A. Aris, L. Babun, E. Tekiner, and A. S. Uluagac, “Minos: A lightweight real-time cryptojacking detection system,” in *NDSS*, 2021.
- [24] W. Wang, B. Ferrell, X. Xu, K. W. Hamlen, and S. Hao, “Seismic: Secure in-lined script monitors for interrupting cryptojacks,” in *Computer Security: 23rd European Symposium on Research in Computer Security, ESORICS 2018, Barcelona, Spain, September 3-7, 2018, Proceedings, Part II 23*, pp. 122–142, Springer, 2018.
- [25] J. D. P. Rodriguez and J. Posegga, “Rapid: Resource and api-based detection against in-browser miners,” in *Proceedings of the 34th Annual Computer Security Applications Conference*, pp. 313–326, 2018.
- [26] A. Kharraz, Z. Ma, P. Murley, C. Lever, J. Mason, A. Miller, N. Borisov, M. Antonakakis, and M. Bailey, “Outguard: Detecting in-browser covert cryptocurrency mining in the wild,” in *The World Wide Web Conference*, pp. 840–852, 2019.
- [27] H. Okhravi, M. Rabe, T. Mayberry, W. Leonard, T. Hobson, D. Bigelow, and W. Streilein, “Survey of cyber moving targets,” *Massachusetts Inst of Technology Lexington Lincoln Lab, No. MIT/LL-TR-1166*, 2013.
- [28] F. B. Cohen, “Operating system protection through program evolution,” *Computers & Security*, vol. 12, no. 6, pp. 565–584, 1993.
- [29] S. Forrest, A. Somayaji, and D. Ackley, “Building diverse computer systems,” in *Proceedings. The Sixth Workshop on Hot Topics in Operating Systems (Cat. No.97TB100133)*, pp. 67–72, 1997.
- [30] M. Eichin and J. Rochlis, “With microscope and tweezers: an analysis of the internet virus of november 1988,” in *Proceedings. 1989 IEEE Symposium on Security and Privacy*, pp. 326–343, 1989.
- [31] J. Cabrera Arteaga, “Artificial software diversification for webassembly,” 2022. QC 20220909.
- [32] A. Rossberg, B. L. Titzer, A. Haas, D. L. Schuff, D. Gohman, L. Wagner, A. Zakai, J. F. Bastien, and M. Holman, “Bringing the web up to speed with webassembly,” *Commun. ACM*, vol. 61, p. 107–115, nov 2018.
- [33] D. Bryant, “Webassembly outside the browser: A new foundation for pervasive computing,” in *Proc. of ICWE 2020*, pp. 9–12, 2020.
- [34] B. Spies and M. Mock, “An evaluation of webassembly in non-web environments,” in *2021 XLVII Latin American Computing Conference (CLEI)*, pp. 1–10, 2021.

- [35] E. Wen and G. Weber, “Wasmachine: Bring iot up to speed with a webassembly os,” in *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pp. 1–4, IEEE, 2020.
- [36] A. Hilbig, D. Lehmann, and M. Pradel, “An empirical study of real-world webassembly binaries: Security, languages, use cases,” *Proceedings of the Web Conference 2021*, 2021.
- [37] L. Wagner, M. Mayer, A. Marino, A. Soldani Nezhad, H. Zwaan, and I. Malavolta, “On the energy consumption and performance of webassembly binaries across programming languages and runtimes in iot,” in *Proceedings of the 27th International Conference on Evaluation and Assessment in Software Engineering, EASE '23*, (New York, NY, USA), p. 72–82, Association for Computing Machinery, 2023.
- [38] B. L. Titzer, “Whose Baseline (compiler) is it anyway?,” *arXiv e-prints*, p. arXiv:2305.13241, May 2023.
- [39] N. Mäkitalo, T. Mikkonen, C. Pautasso, V. Bankowski, P. Daubaris, R. Mikkola, and O. Beletski, “Webassembly modules as lightweight containers for liquid iot applications,” in *International Conference on Web Engineering*, pp. 328–336, Springer, 2021.
- [40] P. K. Gadepalli, S. McBride, G. Peach, L. Cherkasova, and G. Parmer, “Sledge: A serverless-first, light-weight wasm runtime for the edge,” in *Proceedings of the 21st International Middleware Conference*, p. 265–279, 2020.
- [41] I. Bastys, M. Algehed, A. Sjösten, and A. Sabelfeld, “Secwasm: Information flow control for webassembly,” in *Static Analysis* (G. Singh and C. Urban, eds.), (Cham), pp. 74–103, Springer Nature Switzerland, 2022.
- [42] T. Brito, P. Lopes, N. Santos, and J. F. Santos, “Wasmati: An efficient static vulnerability scanner for webassembly,” *Computers & Security*, vol. 118, p. 102745, 2022.
- [43] F. Marques, J. Fragoso Santos, N. Santos, and P. Adão, “Concolic execution for webassembly (artifact),” Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2022.
- [44] C. Watt, J. Renner, N. Popescu, S. Cauligi, and D. Stefan, “Ct-wasm: Type-driven secure cryptography for the web ecosystem,” *Proc. ACM Program. Lang.*, vol. 3, jan 2019.
- [45] R. M. Tsoupidi, M. Balliu, and B. Baudry, “Vivienne: Relational verification of cryptographic implementations in webassembly,” in *2021 IEEE Secure Development Conference (SecDev)*, pp. 94–102, 2021.